AMENDMENTS TO THE CLAIMS

1. (Original) In a communication network, a method of TCP state migration comprising the steps of:

a) establishing a TCP/IP communication session between a client computer and a first server computer, said first server computer part of a plurality of server computers forming a web cluster containing information, said communication session established for the transfer of data contained within said information;

b) handing off said communication session to a selected server computer from said first server computer over a persistent control channel using TCP handoff modules that are dynamically loadable within TCP/IP stacks in operating systems located at both said first server computer and said selected server computer, that implement a TCP handoff protocol that works within kernel levels of an existing TCP/IP protocol; and

c) migrating a first TCP state of said first server computer to said selected server computer, and a second TCP state of said selected server computer to said first server computer over said control channel.

2. (Original) The method as described in Claim 1, wherein said step a) comprises the steps of:

receiving a SYN packet from said client at a first BTCP module located at said first server computer;

sending said SYN packet upstream to a first TCP module located above said first BTCP module in a first operating system of said first server computer;

receiving a first SYN/ACK packet from said first TCP module;

parsing said first initial TCP state from said first SYN/ACK packet, including a first initial sequence number for said first TCP module associated with said TCP/IP communication session;

sending said SYN/ACK packet to said client;

receiving an ACK packet from said client at said first BTCP module;

sending said ACK packet to said first TCP module;

receiving a web request packet associated with said TCP/IP communication session at said first BTCP module at said first server computer;

storing said SYN, ACK and said web request packet at said first server computer.

3. (Original) The method as described in Claim 2, wherein said step b) comprises the steps of:

examining content of said web request packet;

determining which of said plurality of server computers, a selected server computer, can best process said WEB request packet, based on said content;

sending a handoff request from said first BTCP module to a second BTCP module at said selected server computer over said control channel, if said selected server computer is not said first server computer;

including said SYN packet and said ACK packet in said handoff request packet;

changing a first destination IP address of said SYN packet to a second IP address of said selected server computer, at said second BTCP module;

sending said SYN packet to said second TCP module;

receiving a second SYN/ACK packet at said second BTCP module;

parsing said second initial TCP state from said second SYN/ACK packet, including a second initial sequence number, for said second TCP module, that is associated with said TCP/IP communication session;

changing a second destination IP address of said ACM packet to said second IP address, at said second BTCP module;

updating said ACM packet to reflect said second TCP state of said selected server computer in said communication session;

sending said ACM packet that is updated to said second TCP module; and

sending a handoff acknowledgment message to said first BTCP module.

4. (Original) The method as described in Claim 3, wherein step c) comprises the steps of:

monitoring traffic associated with establishing said TCP/IP communication session in step a), at said first BTCP module, to parse a first initial TCP state of said first server computer, said first initial TCP state associated with said TCP/IP communication session; and

migrating said first initial TCP state to said second BTCP module over said control channel by including said first initial TCP state in said handoff request packet, said first initial TCP state including a first sequence number, such that said second BTCP module can calculate said first TCP state for said first server computer in said TCP/IP communication session.

5. (Currently Amended) The method as described in Claim 3, wherein step c) comprises the steps of:

monitoring traffic associated with handing off said TCP/IP communication session ~~in step c)~~, at said second BTCP module, to parse a second initial TCP state of said selected server computer, said second initial TCP state associated with said TCP/IP communication session; and

migrating said second initial TCP state of said selected server computer to said first BTCP module by including said second initial TCP state in said handoff acknowledgment packet, said second initial TCP state including a second initial sequence number, such that said first BTCP module can calculate said second TCP state for said selected server computer in said TCP/IP communication session.

6. (Original) The method as described in Claim 2, comprising the further steps of:

intercepting a connection indication message sent from said first TCP module to an application layer above said first TCP module at a first upper-TCP (UTCP) module, said connection indication message sent by said first TCP module upon establishing said communication session; and

holding said connection indication message at said first UTCP module.

7. (Original) The method as described in Claim 6, wherein said method comprises the further steps of:

sending a reset packet from said first BTCP module upon receiving said handoff acknowledgment packet to said first TCP module;

discarding said connection indication message at said first UTCP module;

receiving incoming data packets from said client at said first BTCP module;

changing said destination addresses of said incoming data packets to said second IP address;

updating sequence numbers and TCP checksum in said data packets to reflect said second TCP state of said selected server computer; and

forwarding said data packets to said selected server computer.

8. (Original) The method as described in Claim 6, comprising the further steps of:

sending notification from said first BTCP module to said first UTCP module to release said connection indication message, if said selected server computer is said first server computer;

sending incoming data packets, including said web request packet, from said client, received at said first BTCP module, upstream.

9. (Original) The method as described in Claim 1, comprising the further step of:

intercepting outgoing response packets from said selected server computer at a second bottom TCP (BTCP) module located at said selected server computer;

changing source addresses of said response packets to a first IP address of said first server computer;

updating sequence numbers and TCP checksum in said response packets to reflect said first TCP state of said first server computer; and

sending said response packets to said client.

10. (Original) The method as described in Claim 1, comprising the further steps of:

monitoring TCP/IP control traffic for said communication session at said second BTCP module;

understanding when said communication session is closed at said second server computer;

sending a termination message to said first server computer over said control channel;

terminating said TCP/IP communication session at said first server computer by terminating a forwarding mode at said first BTCP module; and

freeing data resources associated with said communication session at said first server computer.

11. (Original) In a communication network, a method of TCP state migration comprising the steps of:

a) establishing a TCP/IP communication session between a client computer and a first server computer, said first server computer part of a plurality of server computers forming a web cluster containing information, said communication session established for the transfer of data contained within said information;

b) monitoring traffic associated with establishing said TCP/IP communication session to understand a first initial TCP state of said first server computer associated with said TCP/IP communication session, at a first bottom-TCP (BTCP) module at said first server computer;

c) receiving a web request associated with said TCP/IP communication session at said first BTCP module at said first server computer;

d) examining content of said web request;

e) determining which of said plurality of server computers, a selected server computer, can best process said web request, based on said content;

f) handing off said communication session to said selected server computer from said first server computer over a persistent control channel, if said selected server computer is not said first server computer;

g) monitoring traffic associated with handing off said TCP/IP communication session to understand a second initial TCP state of said selected server computer associated with said TCP/IP communication session, at a second BTCP module at said selected server computer;

h) migrating said first initial TCP state to said selected server computer over said control channel, such that said second BTCP module can calculate a first TCP state for said first server computer in said TCP/IP communication session;

i) sending a second initial TCP state of said selected server computer to said first BTCP module, such that said first BTCP module can calculate a second TCP state for said selected server computer in said TCP/IP communication session;

j) forwarding data packets received at said first BTCP module from said client to said selected server computer, by changing said data packets to reflect said second TCP state and a second IP address of said selected server computer;

k) sending response packets from said selected server computer directly to said client computer by changing said response packets to reflect said first TCP state and a first IP

address of said first server computer; and

l) terminating said TCP/IP communication session at said first server computer when said TCP/IP communication session is closed.

12. (Original) The method as described in Claim 11, wherein said step a) comprises the steps of:

receiving a packet from said client at said first BTCP module;

sending said SYN packet upstream to a first TCP module located above said first BTCP module in a first operating system of said first server computer;

receiving a first SYN/ACK packet from said first TCP module;

parsing said first initial TCP state from said first SYN/ACK packet, including a first initial sequence number for said first TCP module associated with, said TCP/IP communication session;

sending said SYN/ACK packet to said client;

receiving an ACK packet from said client at said first BTCP module;

sending said ACK packet to said first TCP module;

storing said SYN, ACK and said web request at said first server computer.

13. (Original) The method as described in Claim 11, wherein said step e) comprises the steps of:

sending a handoff request packet from said first BTCP module to said second BTCP module over said control channel;

including said SYN packet and said ACK packet in said handoff request packet;

changing a first destination IP address of said SYN packet to a second IP address of said selected server computer, at said second BTCP module;

sending said SYN packet to said second TCP module;

receiving a second SYN/ACK packet at said second BTCP module;

parsing said second initial TCP state from said second SYN/ACK packet, including a second initial sequence number, for said second TCP module, that is associated with said TCP/IP communication session;

changing a second destination IP address of said ACK packet to said second IP address, at said second BTCP module;

updating said ACK packet to reflect said second TCP state of said selected server computer in said communication session;

sending said ACK packet that is updated to said second TCP module; and

sending a handoff acknowledgment message to said first BTCP module.

14. (Original) The method as described in Claim 13, wherein said ACK packet includes said first initial TCP state of said first server computer as provided for in step f).

15. (Original) The method as described in Claim 13, wherein said handoff acknowledgment includes said second initial TCP state of said second server computer, including a second initial sequence number, for said second TCP module, that is associated with said TCP/IP communication session as provided for in step i).

16. (Original) The method as described in Claim 13, comprising the further steps of:

intercepting a connection indication message sent from said first TCP module to an application layer above said first TCP module at a first upper-TCP (UTCP) module, said connection indication message sent by said first TCP module upon establishing said communication session; and

holding said connection indication message at said first UTCP module.

17. (Original) The method as described in Claim 16, wherein step h) comprises the further steps of:

sending a reset packet from said first BTCP module upon receiving said handoff acknowledgment packet to said first TCP module;

discarding said connection indication message at said first UTCP module;

receiving incoming data pa from said client at said first BTCP module;

changing said destination addresses of said incoming data packets to said second IP address;

updating sequence numbers and TCP checksum in said data packets to reflect said second TCP state of said selected server computer; and

forwarding said data packets to said selected server computer.

18. (Original) The method as described in Claim 11, wherein step k) comprises the steps of:

intercepting outgoing response packets from said selected server computer at said second BTCP module;

changing source addresses of said response packets to said first IP address;

updating sequence numbers and TCP checksum in said response packets to reflect said first TCP state of said first server computer; and

sending said updated response packets to said client.

19. (Original) The method as described in Claim 11, wherein step 1) comprises the steps of:

monitoring TCP/IP control traffic for said communication session at said second BTCP module;

understanding when said communication session is closed at said second server computer;

sending a termination message to said first server computer over said control channel;

terminating a forwarding mode at said first BTCP module; and

freeing data resources associated with said communication session at said first server computer.

20. (Original) The method as described in Claim 16, comprising the further steps of:

sending notification from said first BTCP module to said first UTCP module to release said connection indication message, if said selected server computer is said first server computer; and

sending incoming data packets, including said web request, from said client, received at said first BTCP module, upstream.

21. (Original) The method as described in Claim 11, wherein each of said plurality of server computers is constructed similarly including BTCP modules located downstream from TCP modules, and UTCP modules located upstream from TCP modules.

22. (Original) The method as described in Claim 12, comprising the further step of storing said web request, said SYN packet, said ACK packet, and said web request at said first server computer.

23. (Original) The method as described in Claim 22, wherein said control channel allows for communication between all UTCP modules.

24. (Original) The method as described in Claim 11, wherein said plurality of server computers is coupled together over a wide area network in said communication network.

25. (Original) The method as described in Claim 11, wherein said information is partitioned/partially replicated throughout each of said plurality of server computers.

26. (Original) A server computer comprising:

an upper TCP (UTCP) module located above a TCP module in an operating system of said server computer;

a bottom TCP (BTCP) module located below said TCP module, said UTCP, TCP, and BTCP modules implementing a method of handing off a communication session between a first node and second node in a cluster network that works within the kernel level of an existing TCP/IP protocol, by migrating TCP states associated with said first and second nodes.

27. (Original) The server computer as described in Claim 26, wherein said method comprises the steps of:

a) establishing a TCP/IP communication session between a client computer and said server computer, said first node, said server computer part of a plurality of server computers forming said cluster network containing information, said communication session established for the transfer of data contained within said information;

b) receiving a web request associated with said TCP/IP communication session at a first BTCP module at said server computer;

c) examining content of said web request;

d) determining which of said plurality of server computers, a selected server computer, can best process said web request, based on said content;

e) handing off said communication session to said selected server computer from said server computer over a persistent control channel, if said selected server computer is not said server computer; and

f) migrating a first TCP state of said server computer to said selected server computer, and sending a second TCP state of said selected server computer to said server computer over said control channel.

28. (Original) The server computer as described in Claim 27, wherein step a) of said method comprises the steps of:

receiving a SYN packet from said client at said BTCP module;

sending said SYN packet upstream to said TCP module;

receiving a first SYN/ACK packet from said TCP module;

parsing a first initial TCP state from said first SYN/ACK packet, including a first initial sequence number for said TCP module associated with said TCP/IP communication session;

sending said SYN/ACK packet to said client;

receiving an ACK packet from said client at said BTCP module;

sending said ACK packet to said TCP module;

storing said SYN, ACK at said server computer.

29. (Original) The server computer as described in Claim 28, wherein said method comprises the steps of:

sending a handoff request packet from said BTCP module to a second BTCP module over said control channel, said second BTCP module located below a second TCP module in a second operating system at said selected server computer;

including said SYN packet and said ACK packet in said handoff request;

receiving a handoff acknowledgment message at said BTCP module from said second BTCP module.

30. (Original) The server computer as described in Claim 29, wherein said step f) of said method comprises the steps of:

monitoring traffic associated with establishing said TCP/IP communication session in step a), at said BTCP module, to parse a first initial TCP state of said server computer, said first initial TCP state associated with said TCP/IP communication session; and

migrating said first initial TCP state to said second BTCP module over said control channel by including said first initial. TCP state in said handoff request, said first initial TCP state including a first sequence number, such that said second BTCP module can calculate said first TCP state for said server computer in said TCP/IP communication session.

31. (Original) The server computer as described in Claim 29, wherein said method comprises the further steps of:

intercepting a connection indication message sent from said first TCP module to an application layer above said first TCP module at a first upper-TCP (UTCP) module, said connection indication message sent by said first TCP module upon establishing said communication session; and

holding said connection indication message at said first UTCP module.

32. (Original) The computer system as described in Claim 31, wherein said method comprises the further steps of:

sending a reset packet from said first BTCP module upon receiving said handoff acknowledgment packet to said first TCP module;

discarding said connection indication message at said first UTCP module;

receiving incoming data packets from said client at said first BTCP module;

changing said destination addresses of said incoming data packets to said second IP address;

updating sequence numbers and TCP checksum in said data packets to reflect said second TCP state of said selected server computer; and

forwarding said data packets to said selected server computer.

33. (Original) The server computer s described in Claim 31, said method comprising the further steps of:

sending notification from said BTCP module to said UTCP module to release said connection indication message, if said selected server computer is said server computer;

sending incoming data packets, including said web request, from said client, received at said first BTCP module, upstream.

34. (Original) The server computer as described in Claim 26, said method comprising the further steps of:

receiving a handoff request from a first BTCP module located at a first server computer within said cluster network over a persistent control channel, said first server computer having established a communication session with a client computer, said communication session established for the transfer of data contained within said server computer, said handoff request including a SYN packet and an ACK packet, said SYN and ACK packet used for establishing said communication session between said client and said first server computer, said ACK packet including a first initial TCP state of said first server computer in said communication session, including a first initial TCP sequence number;

changing a first destination IP address of said SYN packet to a second IP address of said server computer, at said BTCP module;

sending said SYN packet to said TCP module;

receiving a SYN/ACK packet at said second BTCP module;

parsing a second initial TCP state from second SYN/ACK packet, including a second initial sequence number, for said TCP module, said second initial TCP state associated with a second TCP state for said server computer in said TCP/IP communication session;

changing a second destination IP address of said ACK packet to said second IP address, at said BTCP module;

updating said ACK packet to reflect said second TCP state of said selected server computer in said communication session;

sending said ACK packet that is updated to said TCP module; and

sending a handoff acknowledgment message to said first BTCP module over said control channel.

35. (Original) The server computer as described in Claim 34, wherein said method comprises the further steps of:

monitoring traffic associated with handing off said TCP/IP communication session to said server computer, at said BTCP module, to parse said second initial TCP state of said server computer, said second initial TCP state associated with said TCP/IP communication session; and

sending said second initial TCP state of said server computer to said first BTCP module by including said second initial TCP state in said handoff acknowledgment, said second initial TCP state including a second initial sequence number, such that said first BTCP module can calculate said second TCP state for said server computer in said TCP/IP communication session.

36. (Original) The server computer as described in Claim 34, wherein said method comprises the further steps of:

intercepting outgoing response packets from said server computer at said second BTCP module;

changing source addresses of said response packets to said first IP address;

updating sequence numbers and TCP checksum in said response packets to reflect said first TCP state of said first server computer; and

sending said response packets to said client.

37. (Original) The server computer as described in Claim 34, wherein said method comprises the further steps of:

monitoring TCP/IP control traffic for said communication session at said BTCP module;

understanding when said communication session is closed at said server computer; and

sending a termination message to said first server computer over said control channel.